Technical Note: Parameter Atlases and Function Sculpting in Neural Networks

Motivation

Conventional neural networks determine routing decisions recursively: activations in one layer influence routing decisions in subsequent layers. This produces evolving computational pathways whose structure changes throughout the forward pass.

An alternative approach is to determine all routing decisions at the outset from the input itself using a global locality-preserving sketch, for example via locality-sensitive hashing (LSH).

Core Idea

The input is first transformed into a global routing code:

input
→ global locality-preserving sketch
→ routing decisions for all layers
→ sparse computation

The routing code determines, in advance, which parameters will participate in computation throughout the network.

The selected parameters are then trained normally via gradient descent.

Thus, for a given input, the network effectively plans its computational route before beginning the computation, analogous to selecting a route on a map before a journey.

The Parameter Atlas

Under the manifold hypothesis, natural data are assumed to lie on or near a low-dimensional manifold embedded in a high-dimensional space.

A global routing mechanism may therefore be interpreted as constructing a discrete **Parameter Atlas** of the data manifold.

Each routing code selects one or more local charts:

input x
→ chart C(x)
→ parameter subset P(x)
→ computation

Nearby inputs tend to produce similar routing codes and therefore activate overlapping parameter subsets.

Consequently:

nearby inputs
→ nearby charts
→ similar parameter usage
→ smooth functional variation

The atlas determines *where* computation occurs.

Function Sculpting

Once a chart has been selected by the Parameter Atlas, gradient descent shapes the computation performed within that chart.

This learning process may be viewed as **Function Sculpting**.

The atlas selects the material; learning sculpts the local function.

For a fixed input, the network collapses to an effective linear mapping

$$y = M(x)x$$

where the input-dependent matrix M(x) is assembled from the parameter subsets selected by the atlas.

Learning therefore consists of sculpting a collection of local mappings associated with the charts of the atlas.

Two-System View

The network can be viewed as consisting of two largely independent systems:

1. Parameter Atlas

Responsible for:

- routing;
- chart selection;
- parameter activation;
- preserving locality in computation.

2. Function Sculpting

Responsible for:

- fitting local functions;
- shaping effective mappings;
- adapting selected parameters via gradient descent.

In this view:

> The Parameter Atlas determines where computation takes place, while Function Sculpting determines what computation takes place.

Potential Advantages

- stable and deterministic routing;
- reduced recursive routing dependencies;
- improved parameter utilization;
- smoother locality-preserving computation;
- reduced pathway monopolization;
- sparse specialization with global continuity.

Research Question

An open question is whether a fixed locality-preserving Parameter Atlas combined with gradient-based Function Sculpting can outperform conventional recursively-gated architectures.

If successful, this would suggest that much of the benefit of deep architectures arises not from learning routing itself, but from learning local functions within an appropriately chosen atlas of the data manifold.

Keywords

locality-sensitive hashing, manifold hypothesis, sparse neural networks, conditional computation, mixture of experts, random projections, input-dependent computation, effective linearization, sparse routing.